Soft-decision decoding of RS codes

Justesen, Jørn

Published in:
IEEE International Symposium on Information Theory 2005

Link to article, DOI:
10.1109/ISIT.2005.1523528

Publication date:
2005

Document Version
Publisher's PDF, also known as Version of record

Soft-Decision Decoding of RS Codes

Jorn Justesen
COM, Technical University of Denmark
DK 2800 Kgs Lyngby , Denmark
Email: jju@com.dtu.dk

***Abstract*— By introducing a few simplifying assumptions we derive a simple condition for successful decoding using the Koetter-Vardy algorithm for soft-decision decoding of RS codes. We show that the algorithm has a significant advantage over hard decision decoding when the code rate is low, when two or more sets of received symbols have substantially different reliabilities, or when the number of alternative transmitted symbols is very small.**

I. Introduction

In [1] Koetter and Vardy studied an extension of the Sudan-Guruswami algorithm [2] where the interpolating polynomial is required to have certain multiplicities of zeros for several likely symbols in each position. In particular they analyzed the case where the multiplicities are chosen to be approximately proportional to the conditional probabilities of the symbol values. In general it is difficult to interpret the condition for successful decoding in a simple way. However, by introducing a few simplifying assumptions we derive a much simpler condition for decoding, and in this way we obtain bounds on the performance of the Koetter-Vardy (KV) algorithm. We show that the algorithm has a significant advantage over hard decision decoding when the code rate is low, when two or more sets of received symbols have substantially different reliabilities, or when the correct symbol is on a small list of possible transmitted symbols.

II. The Condition for Successful List Decoding

We consider decoding of an (N, K) RS code over the field $F(q)$. The list decoding algorithm developed in [2] is based on a two-variable interpolating polynomial with the received symbol values as roots of a given multiplicity. In the KV algorithm, the concept is extended by allowing a list of input symbols corresponding to roots of variable multiplicity. When the sum of the multiplicities for each position is upper bounded by a constant, the result of the KV algorithm is a list such that the size is polynomial in N. Decoding is considered to be successful if the transmitted codeword is on the list, and thus all codewords satisfying the condition for successful decoding are found on the list. List decoding allows more than $(N-K)/2$ errors to be decoded, even though the result is not always unique. On the other hand list decoding with a polynomial list is only possible within the Hamming bound, and in most cases the list contains only a single word.

In [1] the input to the decoder is a q by N matrix, Π, called the reliability matrix, which has entries

$$\pi_{ij} = P[a_i \mid y_j]$$

where a_i is a symbol from the code alphabet, A, and y_j is the received symbol, which may belong to a larger alphabet, B. The multiplicities of the zeros corresponding to the input lists are entries in the multiplicity matrix, a q by N matrix, $M = [m_{ij}]$. As discussed in [1], the integer entries in M should be chosen to approximate Π after a suitable normalization.

If the transmitted word is $c = [c(j)]$, where we let $c(j) = i$ indicate that the transmitted symbol is a_i, the condition for successful decoding is

$$\frac{\sum_j m_{c(j),j}}{\sqrt{\sum_{i,j} m_{ij}(m_{ij}+1)}} > \sqrt{K-1} \tag{1}$$

The authors argue that a suitable choice of the multiplicities can be obtained by approximating the conditional probabilities for the q symbols given the received value by a vector of integers. We shall simplify this condition by assuming that the integer entries of M are large enough to allow an accurate approximation to these probabilities, and also neglect smaller terms. Thus by normalizing the multiplicities, we obtain a set of weights, $w_{ij} \approx \pi_{ij}$, and the condition can be expressed in terms of these weights as

$$\frac{\sum_j w_{c(j),j}}{\sqrt{\sum_{i,j} w_{ij}^2}} > \sqrt{K} \tag{2}$$

For each column index, j, the sum of the probabilities is 1, but we allow the normalized sum of the multiplicities to be less than one, since it may be preferable to replace small entries with 0.

III. Quantized Symbols and Typical Error Patterns

We assume that the received symbols belong to a finite alphabet, B, which may be larger than the code alphabet, A. The received symbols would typically be obtained by filtering, sampling, and quantizing a noisy analogue signal, but the exact mechanism is immaterial. Assuming that the transmitted symbols a_i are equally likely, the conditional distribution given a particular received symbol b_j may be written as

$$P[a_i \mid b_j] = kP[b_j \mid a_i]/P[b_j]$$

For each transmitted symbol we assume that there is a unique received symbol, $a_i' \in B' \subset B$, which maximizes $P[b_j \mid a_i]$, and that these symbols are distinct. Often these symbols may be interpreted as noiseless versions of the transmitted symbols. For each received symbol, w_{ij} assumes a

finite set of values, and the condition (2) depends on only this set, the number of times each value occurs, and the weight of the transmitted symbol. Thus for this discussion we can assume that the weights are listed in decreasing order, and we do not need to distinguish symbols with the same list of weights. We are interested in the situation where these lists have few nonzero entries and where the number of distinct distributions is much smaller than the length of the code. The exact probabilities may be approximated to reduce the number of distinct lists, and in particular we assign a probability of zero to all events that have very small probability. The use of such approximated values is justified in part by the need to keep the sum of the multiplicities limited, but also by the observation that the performance of the algorithm is rather insensitive to small changes in the weights.

Definition: An error type associated with a pair of transmitted and received symbols, a_i and b_j, is a list of weights, and an indication of the weight of the symbol actually transmitted.

For binary transmitted symbols and either the binary symmetric channel or a quantized Gaussian channel, this definition agrees with the usual concept of errors. For larger transmitted alphabets there are more error types, and a particular error type can occur only for certain transmitted/received symbols. Nevertheless, our assumption that there are relatively few error types is consistent with typical situations involving modulation formats or inner codes.

As a simple case, let the received alphabet equal the code alphabet. Thus for a given channel, the probability distribution given a received symbol describes the probability that the transmitted symbol is at a certain distance from the one transmitted. If the set of transmitted signals has sufficient symmetry, this distribution may be the same for all symbols, but for modulation formats like QAM, certain symbols have fewer neighbors, and thus there are a number of different probability distributions. Similarly the received symbols can often be divided into a small number of classes, where each class is associated with certain error types.

On the average each symbol appears $NP[b_j]$ times in a received block, and of these cases the transmitted symbols was a_i in $NP[b_j]P[a_i \mid b_j]$ instances. Both of these averages are less than one, but our assumption is that a summation over equivalent symbols gives a moderate value. For (2) to be satisfied in this case we must have

$$\frac{\sum_{i,j} NP[b_j]w_{ij}^2}{\sqrt{\sum_{i,j} NP[b_j]w_{ij}^2}} > \sqrt{K} \tag{3}$$

or

$$\sum_{i,j} NP[b_j]w_{ij}^2 > K/N \tag{4}$$

We can interpret (4) as indicating that the rate of the code must be low enough to allow the condition for decoding to be satisfied for a typical distribution of errors. Based on this observation and the simplified expression, we derive bounds on the performance of the KV algorithm in the next section.

IV. A Bound on Correctable Errors

In this section we derive bounds on performnance of the KV algorithm using (4) as the limiting situation. We then specialize the bounds to some cases that are particularly important in applications. Let r_j be the number of received symbols in a block that belong to a certain class. We can then collect the terms in (2) to obtain the square of the denominator

$$\sum_{i,j} r_j w_{ij}^2$$

This quantity is independent of which symbols were actually transmitted. Thus if (3) is satisfied, and the number of errors of the type characterized by (a_i, b_j) is less than or equal to $r_j w_{ij}$ for all a_i such that $w_{ij} < \max_i\{P[a_i \mid b_j]\}$, the block is correctly decoded, since the numerator is at least as large as in (3). Thus, for a received word with a given symbol distribution, we can interpret (4) as an upper bound on the number of errors of various type that the algorithm will correct. However, in general it is difficult to obtain a bound that does not depend on the composition of the particular received word. Even in the case of only two types of received symbols, the left side of (2) is not always a monotone function of r_1.

In order to get a more useful bound we need the assumption that if (2) is satisfied, it will always remain satisfied if a received symbol is replaced by the corresponding 'correct' symbol a_i'. When a symbol is corrected, the denominator usually also increases, but for most cases of interest, it is clear that the value of the fraction increases. With this assumption let the set of numbers r_j' be chosen such that

$$\sum_j r_j' = N$$

$$\sum_{i,j} r_j' w_{ij} > K/N$$

We then have the following condition for correct decoding of the received block:

Theorem 1: If the number of received symbols of each type $b_j \notin B'$ is $r_j \leq r_j'$, and the number of errors of each type is at most

$$r_j' P[a_i \mid b_j]$$

list decoding by the KV algorithm succeeds.

Proof: Under these assumptions, the left side of (2) is lower bounded by the fraction where r_j is replaced by r_j', since some correct symbols are replaced by less reliable values. We can then use the argument from the beginning of the section, since the composition of the codeword is fixed, and the number of errors is bounded.

In this bound we neglect cases where more errors of one type can be decoded because there are fewer errors of other types. However, if the number of error types increases slowly with the block length, as for example a power of $log(q)$, we get a tight bound for large N.

Under the same assumptions we have

Theorem 2: For the set of error patterns specified in Theorem 1, (4) is an upper bound on the rate of a code that can be decoded by the KV algorithm.

We shall now consider some special cases of Theorems 1 and 2. If r symbols are erased, we simply assign a weight of 0 to all of them. Clearly one value is correct, but for a large alphabet $1/q$ is too small to make a difference. Thus from (4) we find

$$N - r > K \text{ or } r < N - K$$

which just serves as a check on this approach. If a list of n possibilities is given, $n << q$, and each is assigned a probability of $1/n$, we find

$$N - r + r/n > K \text{ or } r(n-1)/n < N - K$$

We may interpret this result as saying that a list of two values counts as half of an erasure, a list of three as $2/3$, etc. Thus very small lists offer an advantage compared to erasures, whereas longer lists are of negligible value. From an information theory point of view we would expect the cost of a binary list to be one bit, but the algorithm is far from this limit. The result can be easily extended to include unequal probabilities for the alternatives.

Consider the case where for a set of received symbols the probability of error is known, but other values each have probabilities that are too small to give a significant contribution. Again as a check, let us first assume that all symbols have error probability p:

$$N(1-p)^2 > K \text{ or } p < 1 - \sqrt{K/N}$$

which is the well-known bound for the Sudan-Guruswami algorithm. If several sets of symbols have different error probabilities, we get from (4)

$$\textstyle\sum_j r_j(1-p_j)^2 > K$$

In particular if r symbols have a low reliability while the rest are more reliable we have

$$(1 - r/N)(1-p_1)^2 + r(1-p_2)^2/N > K/N$$

Thus if we find the rates on the square root bound for the two error probabilities, the rate for a code correcting a mixture of the two probabilities is on a straight line connecting these points. Clearly this means that more errors are corrected than in the case where the average error probability applies to all positions. However, there is only a significant difference if the higher error probability is large.

V. Applications

In this section we consider two typical applications, QAM modulation with a large alphabet and concatenated codes with a binary inner code.

Example 1: QAM can serve as an example of a channel with a large alphabet, but only a small number of likely errors. We consider the simplest case: The alphabet is so large that we can neglect the influence of extreme symbols with fewer neighbors, and the receiver uses hard decisions. If p is the probability of error in one dimension, the probabilities of the 8 closest neighbors are

$$p - 2p^2 \text{ and } p^2$$

We neglect other errors. From (4) we get

$$1 - 8p + 28p^2 - \ldots > K/N$$

where as the standard decoding algorithm gives

$$1 - 8p + 8p^2 > K/N$$

Thus for $p = 1/16$ the rate is improved from $17/32$ to $39/64$.

Example 2: In a concatenated code with an inner block code, the basic RS decoding algorithm requires that the number of errors, t, and the number of erasers, e, from the inner decoder satisfy $2t + e < (N - K)$. Generalized minimum distance decoding ensures that all error patterns of weight less than half the product of the distances are decoded, but this algorithm does not improve the performance much. To get an estimate of the performance of the KV algorithm we consider the PG code $(21, 12, 6)$ for which the necessary details can readily be worked out. Let the average number of bit errors in an inner codeword be 2. It follows from the binomial distribution that the probability of 0 or 1 error in a block is 0.39, and in this case the decision has a high reliability. Two errors are corrected, but the probability of decoding error (if 4 errors actually occur) is 0.12. A small fraction of weight 3 error patterns are uniquely decoded, and for simplicity we merge this set with the double errors. The remaining 1120 weight 3 error patterns are in cosets which give a list of 4 possibilities. The remaining errors of weight 4 are treated as erasures, and in our estimate we neglect the contributions from weight 5 errors. In this way we can apply (4) to get

$$K/N < 0.39 + 0.32(1 - 0.12)^2 + 0.16/4 = 0.68$$

Thus compared to standard errors-and-erasures decoding of the outer code, there is a gain from the small list size of weight 3 errors and also a small gain associated with distinguishing the different reliabilities.

VI. Conclusion

Using a simplified expression for the decoding criterion we have derived a bound on the error patterns that can be decoded by the Koetter-Vardy list decoding algorithm. As demonstrated in several specific examples, the improvements are significant only for fairly low rates and short lists of input symbols with high conditional probabilities.

References

[1] R. Koetter and A. Vardy, "Algebraic soft-decision decoding of Reed-Solomon codes," *IEEE Trans. Inform. Theory*, vol. 49, pp. 2809–2825, Nov. 2003.

[2] V. Guruswami and M. Sudan, "Improved decoding of Reed-Solomon and algebraic-geometric codes," *IEEE Trans. Inform. Theory*, vol. 45, pp. 1757–1767, Sept. 1999.